PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G07F 7/08, G07B 15/00 — A1

(11) International Publication Number: WO 99/38129

(43) International Publication Date: 29 July 1999 (29.07.99)

(21) International Application Number: PCT/US99/01388

(22) International Filing Date: 21 January 1999 (21.01.99)

(30) Priority Data:
09/012,750 23 January 1998 (23.01.98) US

(71) Applicant (*for all designated States except US*): AMERICAN EXPRESS TRAVEL RELATED SERVICES COMPANY, INC. [US/US]; American Express Tower, World Financial Center, New York, NY 10285 (US).

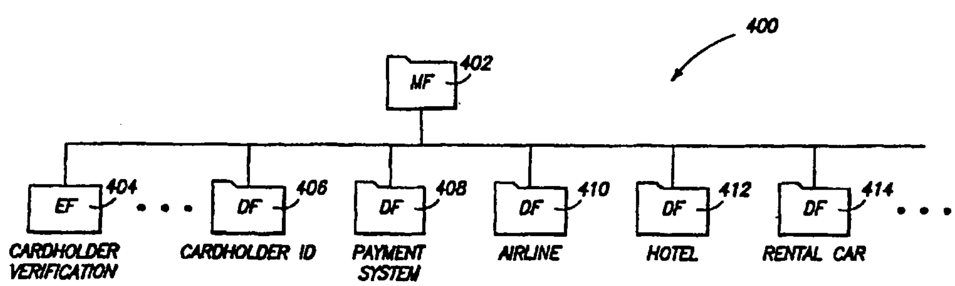
(72) Inventors; and
(75) Inventors/Applicants (*for US only*): HOHLE, William [US/US]; 10882 West Cedar Fort Road, Lehi, UT 84043 (US). PETIT, Frederic [US/US]; 8885 South Goldwater Drive #2197, Sandy, UT 84094 (US).

(74) Agent: POTE, Daniel, R.; Snell & Wilmer L.L.P., One Arizona Center, 400 East Van Buren, Phoenix, AZ 85004-0001 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: METHODS AND APPARATUS FOR A TRAVEL-RELATED MULTI-FUNCTION SMARTCARD



(57) Abstract

Methods and apparatus for a smartcard system are provided which securely and conveniently integrate important travel-related applications. In one embodiment, a smartcard system includes a cardholder identification application and various additional applications useful in particular travel contexts; for example, airline, hotel, rental car, and payment-related applications. Furthermore, memory space and security features within specific applications provide partnering organizations (e.g., airlines, hotel chains, and rental car agencies) the ability to construct custom and secure file structures.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav	TM	Turkmenistan
BF	Burkina Faso	GR	Greece		Republic of Macedonia	TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's	NZ	New Zealand		
CM	Cameroon		Republic of Korea	PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

METHODS AND APPARATUS FOR A TRAVEL-RELATED MULTI-FUNCTION SMARTCARD

Technical Field

The present invention relates generally to the use of integrated circuit cards, or "smartcards," for commercial transactions and, more particularly, to methods and apparatus for conveniently storing, retrieving, and updating data related to a cardholder's travel information in the context of a distributed transaction system.

Background Art and Technical Problems

Despite advances in information technology and process streamlining with respect to travel arrangements, the modern traveler is often subjected to unnecessary delays, petty inconveniences, and oppressive paperwork. These travel burdens are most evident in the airline, hotel, and rental car industries, where arranging and paying for services and accommodations can involve significant time delays due to miscommunication, poor record-keeping, and a host of other administrative inefficiencies.

Smartcard technology, as described below, has had limited success in addressing some of these problems. The term "smartcard" refers generally to wallet-sized or smaller cards incorporating a microprocessor or microcontroller to store and manage data within the card. More complex than magnetic-stripe and stored-value cards, smartcards are characterized by sophisticated memory management and security features. A typical smartcard includes a microcontroller embedded within the card plastic which is electrically connected to an array of external contacts provided on the card exterior. A smartcard microcontroller generally includes an electrically-erasable and programmable read only memory (EEPROM) for storing user data, random access memory (RAM) for scratch storage, and read only memory (ROM) for storing the card operating system. Relatively simple microcontrollers are adequate to control these functions. Thus, it is not unusual for smartcards to utilize 8-bit,

5 MHZ microcontrollers with about 8K of EEPROM memory (for example, the Motorola 6805 or Intel 8051 microcontrollers).

A number of standards have been developed to address general aspects of integrated circuit cards, e.g.: *ISO 7816-1, Part 1: Physical characteristics* (1987); *ISO 7816-2, Part 2: Dimensions and location of the contacts* (1988); *ISO 7816-3, Part 3: Electronic signals and transmission protocols* (1989, Amd. 1 1992, Amd. 2 1994); *ISO 7816-4, Part 4: Inter-industry commands for interchange* (1995); *ISO 7816-5, Part 5 Numbering system and registration procedure for application identifiers* (1994, Amd. 1 1995); *ISO/IEC DIS 7816-6, Inter-industry data elements* (1995); *ISO/IEC WD 7816-7, Part 7: Enhanced inter-industry commands* (1995); and *ISO/IEC WD 7816-8, Part 8: Inter-industry security architecture* (1995). These standards are hereby incorporated by reference. Furthermore, general information regarding magnetic stripe cards and chip cards can be found in a number of standard texts, e.g., Zoreda & Oton, SMART CARDS (1994), and Rankl & Effing, SMART CARD HANDBOOK (1997), the contents of which are hereby incorporated by reference.

Various attempts have been made to alleviate travel-related inconveniences through the use of smartcard technology. In 1995, for example, the U.S. airline industry led an effort to reduce ticket distribution costs by developing standards for "ticketless travel." Soon thereafter, a joint conference of IATA and ATA adopted a set of specifications entitled *Specifications for Airline Industry Integrated Circuit Cards* (hereinafter, "IATA standard"). Similarly, in the field of financial payment systems, a standard has been developed entitled *EMV Version 2.0, Integrated Circuit Card Specifications for Payment Systems, Parts 1-3* (1995). Both of these specifications are hereby incorporated by reference.

Notwithstanding widespread promulgation of these standards, smartcard efforts tend to remain fragmented, and the resultant benefit to consumers -- particularly consumers who travel -- has been quite minimal. One recent study estimates that approximately nine million smartcards were issued in the transportation and travel industry in 1996, yet, for the most part, these cards remain incompatible; that is, due to differing file structures and/or communication protocols employed, card data

typically can not easily be shared across applications or between industry participants.

Systems and methods are therefore needed in order to overcome these and other shortcomings in the prior art.

Summary of the Invention

The present invention provides methods and apparatus for a smartcard system which securely and conveniently integrates important travel-related applications, thereby overcoming the limitations of the prior art. In accordance with one aspect of the present invention, a smartcard system comprises a cardholder identification application and various additional applications useful in particular travel contexts; for example, airline, hotel, rental car, and payment-related applications. In accordance with another aspect of the present invention, a smartcard system further comprises space and security features within specific applications which provide partnering organizations the ability to construct custom and secure file structures.

Brief Description of the Drawing Figures

The present invention will hereinafter be described in conjunction with the appended drawing figures, wherein like numerals denote like elements, and:

Figure 1 illustrates an exemplary smartcard apparatus;

Figure 2 is a schematic diagram of an exemplary smartcard integrated circuit, showing various functional blocks;

Figure 3 is an exemplary diagram of files and directories arranged in a typical tree structure;

Figure 4 sets forth an exemplary database structure in accordance with a preferred embodiment of the present invention;

Figure 5 sets forth a preferred cardholder ID data structure in accordance with the present invention;

Figure 6 sets forth a preferred payment system data structure in accordance with the present invention;

Figure 7 sets forth a preferred airline data structure in accordance with the present invention;

Figure 8 sets forth a preferred rental car data structure in accordance with the present invention;

Figure 9 sets forth a preferred hotel system data structure in accordance with the present invention; and

Figure 10 illustrates an exemplary distributed transaction system useful in practicing the present invention.

Detailed Description of Preferred Exemplary Embodiments

Referring now to **Figures 1 and 2**, an exemplary smartcard system suitable for practicing the present invention will now be described. A smartcard 100 generally comprises a card body 102 having a communication region 104 for providing contact or non-contact communication between an external device (e.g., a card reader) and an integrated circuit 110 encapsulated within card body 102. Communication region 104 preferably comprises six conductive pads 106 whose placement and size conform to ISO7816-2. More particularly, a communication region 104 in conformance with ISO-7816-2 preferably comprises VCC contact 106(a) (power supply), RST contact 106(b) (reset), CLK contact 106(c) (external clock), GND Contact 106(d) (ground), VPP contact 106(e) (programming voltage), and I/O contact 106(f) (data line).

VCC 106(a) suitably provides power to IC 110 (typically 5.0 V +/- 10%). CLK 106(c) is suitably used to provide an external clock source which acts as a data transmission reference. RST 106(b) is suitably used to transmit a reset signal to IC 110 during the booting sequence. VPP contact 106(e) may be used for programming of EEPROM 212 in IC 110. As is known in the art, however, this contact is generally not used since modern ICs typically incorporate a charge pump suitable for EEPROM programming which takes its power from the supply voltage (VCC 106(a)). I/O 106(f) suitably provides a line for serial data communication with an external device, and GND 106(d) is suitably used to provide a ground reference. Encapsulated integrated circuit 110 is configured to communicate electrically with contacts 106 via any number of known packaging techniques, including, for example, thermosonically-bonded gold wires, tape automated bonding (TAB), and the like.

While an exemplary smartcard is discussed above in the context of a plurality of external contacts, it will be appreciated that contactless cards may also be utilized to practice this invention. That is, non-contact communication methods may be employed using such techniques as capacitive coupling, inductive coupling, and the like. As is known in the art, capacitive coupling involves incorporating capacitive plates into the card body such that data transfer with a card reader is provided through symmetric pairs of coupled surfaces, wherein capacitance values are typically 10-50 picofarads, and the working range is typically less than one millimeter. Inductive coupling employs coupling elements, or conductive loops, disposed in a weakly-coupled transformer configuration employing phase, frequency, or amplitude modulation. In this regard, it will be appreciated that the location of communication region 104 disposed on or within card 100 may vary depending on card configuration. For additional information regarding non-contact techniques, see, for example, contactless card standards ISO/IEC 10536 and ISO/IEC 14443, which are hereby incorporated by reference.

Smartcard body 102 is preferably manufactured from a sufficiently rigid material which is resistant to various environmental factors, e.g., physical deterioration, thermal extremes, and ESD (electrostatic discharge). Materials suitable in the context of the present invention include, for example, PVC (polyvinyl chloride), ABS (acrylonitrile-butadiene-styrol), PET (polyethylene terephthalate), or the like. In a preferred embodiment, chip card 100 conforms to the mechanical requirements set forth in ISO 7810, 7813, and 7816. Body 102 may comprise a variety of shapes, for example, the rectangular ID-1, ID-00, or ID-000 dimensions set forth in ISO-7810. In a preferred embodiment, body 102 is roughly the size and shape of a common credit card and substantially conforms to the ID-1 specification.

Referring now to **Figure 2**, IC 110 preferably comprises regions for Random Access Memory (RAM) 216, Read-Only Memory (ROM) 214, Central Processing Unit (CPU) 202, data bus 210, Input/Output (I/O) 208 and Electrically-Erasable and Programmable Read Only Memory (EEPROM) 212.

RAM 216 comprises volatile memory which is used by the card primarily for scratch memory, e.g., to store intermediate calculation results and data encryption processes. RAM 216 preferably comprises at least 256 bytes.

EEPROM 212 provides a non-volatile memory region which is erasable and rewritable electrically, and which is used to store, *inter alia*, user data, system data and application files. In the context of the present invention, EEPROM 212 is suitably used to store a plurality of files related to cardholder travel information (discussed in greater detail below in conjunction with **Figure 3**). EEPROM 212 preferably comprises at least 8K bytes.

In a preferred embodiment, CPU 202 implements the instruction set stored in ROM 202, handles memory management (i.e., RAM 216 and EEPROM 212), and coordinates input/output activities (i.e., I/O 208).

ROM 214 preferably contains, or is "masked" with, the smart card operating system (SCOS). That is, the SCOS is preferably implemented as hard-wired logic in ROM 214 using standard mask design and semiconductor processing methods well known in the art (e.g., photolithography, diffusion, oxidation, ion implantation, etc.). Accordingly, ROM 214 cannot generally be altered after fabrication. The purpose of such an implementation is to take advantage of the fast access times provided by masked ROMs. ROM 214 suitably comprises about 4K-20K bytes of memory, preferably at least 16K bytes. In this regard, it will be appreciated that alternate memory devices may be used in place of ROM 214. Indeed, as semiconductor technology progresses, it may be advantageous to employ more compact forms of memory, for example, flash-EEPROMs.

The SCOS controls information flow to and from the card, and more particularly facilitates storage and retrieval of data stored within EEPROM 212. As with any operating system, the SCOS operates according to a well-defined command set. In this regard, a variety of known smart card operating systems are suitable for the purpose of this invention, for example, IBM's Multi-Function Card (MFC) Operating System 3.51, the specification of which is hereby incorporated by reference. While the IBM MFC operating system employs the standard tree structure of files and directories substantially in accordance with ISO7816-4 (as detailed below), it will be

appreciated by those skilled in the art that other operating system models would be equally suitable for implementation of the present invention. Moreover, it may be advantageous to allow certain aspects of operating system functionality to exist outside the card, i.e., in the form of blocks of executable code which can be downloaded and executed by the smartcard during a transaction (for example, Java applets, ActiveX objects, and the like).

Given the general characteristics of smartcard 100 as outlined above, it will be apparent that a wide range of microcontrollers and contact-based smartcard products known in the art may be used to implement various embodiments of the present invention. Suitable smartcards include, for example, the model ST16SF48 card, manufactured by SGS-Thomson Microelectronics, which incorporates a Motorola 6805 microcontroller with 16K ROM, 8K EEPROM, and 384 bytes of RAM. It will be appreciated, however, that particular embodiments of the present invention might require more advanced microcontrollers with greater EEPROM capacity (i.e., in the range of about 12-16K). Such systems are well known in the art.

Having thus described an exemplary smartcard 100 and IC 110, an overview of a smartcard file structure in accordance with the present invention will now be described. Referring now to **Figure 4**, file structure 400 is preferably used to store information related to card-holder preferences and various data useful for securing and paying for air travel, rental cars, hotel reservations and the like. More particularly, file structure 400 preferably comprises cardholder ID application 406, payment system application 408, airline application 410, hotel system application 412, rental car application 414, and cardholder verification data 404. It will be appreciated by those skilled in the art that the term "application" in this context refers to self-contained regions of data all directed at a particular function (e.g., airline, hotel, etc.) rather than a block of executable software code, although the use of executable modules as part of any particular application falls within the scope of the present invention.

Cardholder verification data 404 preferably houses data useful in verifying cardholder identity during a transaction. In a preferred embodiment, cardholder

verification data 404 comprises two eight-byte cardholder verification numbers (i.e., PIN numbers) referred to as CHV1 and CHV2.

Cardholder ID application 406 suitably comprises various files related to personal information of the cardholder (e.g., name, addresses, payment cards, driver's license, personal preferences and the like). Cardholder ID application 406 is described in greater detail below in conjunction with **Figure 5.**

Payment system application 408 suitably comprises information useful in effecting commercial transactions, e.g., account number and expiration date information traditionally stored on a magnetic-stripe credit card. Alternatively, Payment system application 408 comprises a full EMV-compliant application suitable for a wide range of financial transactions. Payment system application 408 is described further below in conjunction with **Figure 6.**

Airline application 410 suitably comprises data helpful in streamlining commercial airline travel; for example, relevant personal preferences, electronic tickets, and frequent flier information. Airline application 410 is discussed in greater detail below in conjunction with **Figure 7.**

Hotel application 412 suitably comprises information useful for securing and paying for hotel reservations, including an array of information and preferences associated with a list of preferred hotels as well space for electronic keys. Hotel application 412 is discussed in greater detail below in conjunction with **Figure 9.**

Rental car application 414 suitably comprises data useful in expediting the process of car rental and return, including, for example, car preference and frequent rental information. Rental car application 414 is described in further detail below in conjunction with **Figure 8.**

In each of the above mentioned applications, sophisticated access and encryption schemes are preferably utilized in order to allow multiple parties to make use of certain file structures while preventing unauthorized entry into others. More specifically, partnering organizations (e.g., hotel chains, airlines, and rental car agencies) may create their own tailor-made file structures (i.e., "partner file structures") within card 100. Details of the various security measures employed are described in further detail below in conjunction with **Table 40.**

Referring now to **Figure 10**, smartcard 100 is suitably used in the context of a distributed transaction system. Briefly, cardholder's may employ smartcard 100 at various access points 15 which are connected via network 19 to an issuer 10 and at least one partnering organization 12. Issuer 10 suitably comprises various hardware and software components suitable for client host communications as well as a database system 11. In this context, the term 'issuer' refers to the organization that actually issues the smartcard and retains some high-level access to certain areas of file structure 400 (detailed below).

Partnering organizations 12(a), 12(b), and so on, comprise the various hotel chains, rental-car agencies, airlines, and the like, who have access to appropriate data regions within smartcard 100. Each partnering organization 12 suitably comprises a database 13 and appropriate hardware and software components necessary for completing a transaction over network 19. Network 19 may comprise one or more communication modes, e.g., the public switched telephone network (PSTN), the Internet, digital and analog wireless networks, and the like.

Each access point 15 suitably comprises an appropriate card reader for interfacing with smartcard 100 as well as hardware and software suitable for interfacing with a cardholder and performing a transaction over network 19. Access points 15 are preferably located in areas providing convenient access for traveling cardholder's or cardholder's preparing travel arrangements. Such access points 15 may be located, for example, in airline ticketing and gate areas, rental car facilities, hotel lobbies, travel agencies, and stand-alone kiosks in malls. In addition, businesses might see fit to host an access point 15 to streamline their employees' business travel. Furthermore, an individual cardholder might configure his or her personal computer to act as an access point using appropriate software and peripheral hardware.

In a preferred embodiment of the present invention, data files and directories are stored in a "tree" structure as illustrated in **Figure 3**. That is, the smartcard file structure resembles the well known MS-DOS (Microsoft Disk Operating System) file structure wherein files are logically organized within a hierarchy of directories. Specifically, three types of files are defined in ISO 7816-4: dedicated files (DF),

elementary files (EF), and a master file (MF). The master file is analogous to the MS-DOS "root" directory, and contains all other files and directories. Dedicated files are actually directories or "folders" for holding other DFs or EFs. Thus, MF 302 may contain an arbitrary number of DFs 306, and these DFs (e.g., DF 306(a)) may or may not contain other DFs (e.g., DF 308). Elementary files are used to store user data, and may exist within a dedicated file (e.g., EF 310 within DF 306(a)), or within the master file (e.g., EF 304 within MF 302). Higher level DFs (i.e., DFs which house particular applications) are often referred to as application dedicated files (ADFs).

The MF and each of the DFs and EFs are assigned a unique two-byte file identifier (FID). By convention, the MF is traditionally assigned an FID of '3F00' hex. Selection of an EF or DF by the operating system may then be performed by tracing its entire path starting at the MF. Thus, if the MF contains a DF with a FID 'A100', and this DF in turn contains an EF with a FID 'A101', then this EF could be referenced absolutely by successive selection of FIDs 3F00, A100, and A101. It will be appreciated that the FID is essentially a file name used by the operating system to select directories and files; it is not intended to indicate a physical address within EEPROM 212. As will be appreciated by those skilled in the art, low-level EEPROM addressing is preferably handled by the SCOS in conjunction with CPU 202.

Each file preferably has an associated file header containing various indicia of the particular EF, DF, or MF. More particularly, the file header associated with a particular file preferably includes the file identifier (FID), file size, access conditions, and file structure. In this regard, smartcard 100 suitably employs one of four file structures: transparent, linear fixed, linear variable, or cyclic. For the sake completeness, the nature of these file structures will be briefly reviewed.

A transparent file structure consists of a string of bytes accessed by specifying an offset and byte count. For example, with reference to **Table 1** below, given a *n*-byte string of data, bytes 7 through 10 would be accessed using an offset of six and a length of four.

byte#

1	2	3	4	5	6	7	8	9	10	11	12	13	14	...	...	n

-----------------offset--------------> <-------- length -------->

Table 1: Transparent file structure

A linear fixed file structure comprises a plurality of records of equal length (e.g., a list of phone numbers), wherein access to an individual record is achieved through reference to a record number. In addition, it is possible to refer to the 'next' or 'previous' record relative to the 'current' record (i.e., the most recently accessed record). In contrast, a linear variable file structure comprises records of arbitrary but known length, and is therefore typically more compact than linear fixed data structures.

A cyclic file structure is a type of linear fixed file wherein a pointer is used to point to the last data set written to. After the last data record is written to, the pointer returns to the first record. That is, a cyclic file comprises a series of records arranged in a 'ring'.

A data structure particularly important with regard to storing records as well as secure messaging in smartcard applications is the BER tag-length-value or "TLV" structure in accordance with ISO/IEC 8825, hereby incorporated by reference. In a TLV object, information regarding the type and length of the information is included along with the actual data. Thus, a TLV object comprises a tag which identifies the type of data (as called out by the appropriate specification), a length field which indicates the length in bytes of the data to follow, and a value field, which comprises the primary data. For example, the TLV object illustrated in **Table 2** below encodes the text "phoenix", which has a length of 7 bytes, and corresponds to a the "city" tag of '8C' hex (a hypothetical tag designation).

Tag	Length	Value						
'8C'	'07'	p	h	o	e	n	i	x

Table 2: Exemplary primitive TLV object

It will be appreciated that the meaning of the various tag values must be known to the system *a priori*. That is, in order for the tag field to be useful, the smartcard and any external systems communicating with the smartcard must conform to the same tag specification. In this regard, ISO/IEC 7816-6 defines a series of tags useful in the context of the present invention, as does the IBM MFC 3.2 specification. ISO/IEC 8825 sets forth the basic encoding rules for a TLV system and defines a "template" data object which can be used as a container for multiple TLV objects. That is, it is often advantageous to encapsulate primitive TLV objects within a larger template which is itself a TLV object.

Referring now to **Figure 4**, a preferred smartcard data structure in accordance with the present invention will now be described in detail. Data structure 400 preferably comprises a MF 402 and five DFs: Cardholder ID application 406, Payment system application 408, Airline application 410, Hotel application 412, and Rental car application 414.

In the detailed description to follow, various acronyms and abbreviations will be used to refer to particular data types, formats, and the like. A key to these acronyms and abbreviations is presented in **Table 3** below.

AN	Alphanumeric
N	Numeric
B	Boolean
C	Convention
M	Matrix
D	Data
AR	Bits array
BIN	Binary
RJ	Right-justified
LJ	Left-justified
BCD	Binary coded decimal

Table 3: Key to acronyms

In the discussion that follows, the various features of a preferred data structure are in some cases described using particular file structure types (i.e., transparent, fixed, etc.). Those skilled in the art will realize, however, that any of the common smartcard file structure types are typically suitable for implementing any particular data structure. For example, when a file structure is described as including "a plurality of records," it will be understood that such a structure may be designed, for example, using a list of records assembled in a linear fixed file wherein each record is itself a transparent file (and offset values correspond to the various fields). Alternatively, such a structure may be designed using TLV strings assembled in a linear fixed file or within a larger template TLV. This is the case notwithstanding the fact that particular tag values -- which are for the most part arbitrary -- are not explicitly listed in the tables that follow.

Cardholder ID Application

Referring now to **Figure 5**, Cardholder ID application 406 is used to store various information related to the cardholder. Portions of this information are freely available to the partnering organizations, thereby preventing the storage of redundant information.

More particularly, cardholder ID application 406 preferably comprises directory EF 532, holder_ID DF 502 and miscellaneous DF 530. Holder_ID DF 502 preferably comprises ID EF 504, home EF 506, business EF 508, preferences EF 514, passport

EF 516, authentication EF 520, biometric EF 522, and driver EF 518. Miscellaneous EF 530 preferably comprises payment card EF 510, sequence EF 512, issuance EF 511, preferred programs EF 528, and card number EF 526. These files and their respective functions are discussed in detail below.

Directory EF 532 provides a list of application identifiers and labels for the various high-level DF's existing under cardholder ID application 406. That is, this file serves the function of a high-level directory listing which specifies the location (i.e., FID) and application label for each DF -- in this case, holder_ID DF 502 and miscellaneous DF 530. In a particularly preferred embodiment, directory EF 532 is structured in accordance with EMV 3.0 as shown in **Table 4** below. Preferably, each major application (e.g., hotel, airline, etc.) has an associated directory file with a substantially same file structure.

Record description	External format		Internal format(bytes)	
	Size	Type	Size	Type
Application ID for holder_ID DF	16	AN	16	ASCII
Application label	16	AN	16	ASCII
Application ID for miscellaneous DF	16	AN	16	ASCII
Application label	16	AN	16	ASCII

Table 4: Exemplary cardholder ID directory EF

ID EF 504 preferably includes personal information related to the cardholder, e.g., name, date of birth, emergency contact, general preferences, and the like. In a particularly preferred embodiment, member EF 504 comprises the fields set forth in **Table 5** below. Italicized field names indicate a subcategory within a particular field.

Record d scription	External format		Internal f rmat(bytes)	
	Size	Type	Size	Type
Last Name	30	AN	30	ASCII
First Name	20	AN	20	ASCII
Middle Name	8	AN	8	ASCII
Honorary Title	8	AN	8	ASCII
Name Suffix	4	AN	4	ASCII
Date of Birth	8	D	4	BCD
Social Security Number	10	AN	10	ASCII
Emergency Contact				
Last Name	*20*	*AN*	*20*	*ASCII*
First Name	*10*	*AN*	*10*	*ASCII*
Relation	*1*	*C*	*1*	*BIN*
Phone	*20*	*N*	*10*	*BCD*
Gender	1	AN	1	ASCII
Special Personal Requirements	12	AN	12	M
Language Preference (ISO 639)	2	C	2	ASCII

Table 5: Exemplary ID EF data structure

In the above table, and the tables to follow, both internal and external data formats are listed. As the conservation of EEPROM space is of paramount importance, the "internal" format of data (i.e., within EEPROM 212) may be different from the "external" format of the data (i.e., as read by the card reader at an access point 15). Thus, for example, a date field might consist of a four-byte BCD record within the card, but upon reading and processing by the terminal, this data might be converted to an eight-byte decimal value for more convenient processing.

Home EF 506 preferably includes data related to one or more of the cardholder's home addresses. In a particularly preferred embodiment, home EF 506 comprising the fields set forth in **Table 6** below. The personal travel charge account pointer is preferably used to designate a preferred payment card, and consists of a

number corresponding to one of the payment card records within payment card EF 510 (detailed below).

Record description	External format		Internal format(bytes)	
	Size	Type	Size	Type
Home Address 1	40	AN	40	ASCII
Home Address 2	40	AN	40	ASCII
Home Address City	25	AN	25	ASCII
Home Address State	5	AN	5	ASCII
Home Country (ISO 3166)	2	AN	2	ASCII
Home Address Zip Code	10	AN	10	ASCII
Home Address Telephone	20	N	10	BCD
Home Address FAX	20	N	10	BCD
Home E-mail address	40	AN	40	ASCII
Personal travel charge account number pointer	2	N	1	BCD

Table 6: Exemplary home EF file structure

Business EF 508 preferably includes various data related to the cardholder's business (i.e., addresses, phone numbers, and the like). In a particularly preferred embodiment, business EF 508 comprising the fields set forth in **Table 7** below. In this regard, the credit card pointer field is preferably used to point to a payment card record within payment card EF 510 (detailed below). The cost center, dept., division, and employee ID fields are employer-specific, and may or may not apply in a given case.

Record description	External format		Internal format(byt s)	
	Size	Type	Size	Type
Business Address 1	40	AN	40	ACSII
Business Address 2	40	AN	40	ASCII
Business Address City	25	AN	25	ASCII
Business Address State	5	AN	5	ASCII
Business Country (ISO 3166)	2	AN	2	ASCII
Business Address Zip Code	10	AN	10	ASCII
Business Telephone No.	20	N	10	BCD
Business Address Fax	20	N	10	BCD
Business E-mail Address	40	AN	40	ASCII
Professional Title	10	AN	10	ASCII
Employee ID	10	AN	10	ASCII
Division	20	AN	20	ASCII
Dept	20	AN	20	ASCII
Cost Center	12	AN	12	ASCII
Professional travel account number pointer	2	N	2	BCD
Professional license data	20	AN	20	ASCII
Credit Card pointer	2	N	1	BCD
Company Name	20	AN	20	ASCII

Table 7: Exemplary business EF file structure

Preferences EF 514 preferably comprises data related to the cardholder's default personal preferences. In a particularly preferred embodiment, preferences EF 514 includes a field comprising an array of preferences as set forth in **Table 8** below. Preference values are preferably chosen from a list of preference tags as set forth in **Table 39.**

Record description	External format		Internal f rmat(byt s)	
	Size	Type	Size	Type
Preferences Array	20	C	20	C

Table 8: Examplary preferences EF file structure

Passport EF 516 is preferably used to store cardholder passport information. In a particularly preferred embodiment, passport EF 516 comprises the fields set forth in **Table 9** below.

Record description	External format		Internal format(bytes)	
	Size	Type	Size	Type
Passport Number	20	AN	20	ASCII
Passport Country -- ISO 3166	2	AN	2	ASCII
Issuance Date	8	D	4	BCD
City of Issuance	20	AN	20	AN
Expiration Date	8	D	4	BCD

Table 9: Exemplary passport EF file structure

Driver EF 516 preferably comprises cardholder driver license data. In a particularly preferred embodiment, driver EF 518 comprising the fields set forth in **Table 10** below.

Record descripti n	External format		Internal format(bytes)	
	Size	Type	Size	Type
Driver's License No.	20	a	20	ASCII
Driver's License Issuing State/Country	2	a	2	BCD
License Expiration Date	8	D	4	ASCII
License Type	2	C	4	BCD

Table 10: Exemplary driver EF file structure

Biometric EF 522 is used to store biometric data (preferably encoded) such as fingerprint data, retina scan data, or any other sufficiently unique indicia the cardholder's physical or behavioral characteristics. In a particularly preferred embodiment, biometric EF 522 comprises a single data string as set forth in **Table 11** below.

Record description	External format		Internal format (bytes)	
	Size	Type	Size	Type
Biometrics template	100	AN	100	BIN

Table 11: Exemplary biometric EF file structure

Authentication EF 520 preferably comprises information for static authentication of the cardholder ID 406 application. This data is unique for each card, and is sufficiently complex such that counterfeit values cannot feasibly be created. This prevents creation of "new" counterfeit cards (i.e., cards with new authentication data), but does not prevent creation of multiple copies of the current card.

In a particularly preferred embodiment, authentication EF 520 includes public key certificate fields as shown in **Table 12** below, wherein the external format is

identical to the internal format. Preferably, the issuer RSA key is 640 bits long, and the CA key is 768 bits long.

Record description	Internal format(bytes)	
	Size	Type
Signed Static Application Data	80	B
Static Data Authentication Tag List	16	B
Issuer Public Key Certificate	96	B
Issuer Public Key Exponent	1	B
Issuer Public Key Remainder	20	B

Table 12: Exemplary authentication EF

Turning now to files under miscellaneous DF 530, preferred programs EF 528 preferably comprises data related to the cardholder's preferences as to airline companies, hotels, and rental car agencies. Specifically, this EF, in a particularly preferred embodiment, comprises a plurality of records (e.g., three) indicating preferred companies for each type of travel partner as shown in **Table 13.** The actual data values conform to an arbitrary convention; that is, each airline, hotel, and rental car agency is assigned an arbitrary three-byte code.

Record description	External format		Internal format(bytes)	
	Size	Type	Size	Type
Preferred Airlines	9 (3x3)	C	9	C
Preferred Hotels	9	C	9	C
Preferred Rental Cars	9	C	9	C

Table 13: Exemplary programs EF

Payment card EF 510 is preferably used to catalog information related to the cardholder's various payment cards, i.e., debit cards, charge cards, and the like. In

a particularly preferred embodiment, payment card EF comprises card numbers and expiration dates for two cards as shown in **Table 14**. The "ISO" and "non-ISO" designations refer to ISO-7813, which specifies a particular payment card number format. Thus, in a preferred embodiment, either an ISO or non-ISO card number scheme may be used. Moreover, it will be appreciated that this data set is sufficient only for "card not present" transactions, for example, transactions taking place remotely where only the card number and expiration date are required to effect a transaction. Data stored within payment system application 408 (described below) must be used to effect a "card present" transaction.

Record description	**External format**		**Internal format(bytes)**	
	Size	Type	Size	Type
First Payment Card # (ISO)	19	N	10	BCD
First Payment Card Expiration Date	8	D	4	BCD
Second Payment Card # (non-ISO)	20	AN	20	ASCII
Second Payment Card Expiration Date	8	D	4	BCD

Table 14: Exemplary payment card EF file structure

Sequence EF 512 preferably includes information used to provide synchronization of the host and smartcard databases. In a particularly preferred embodiment, sequence EF 512 comprises a plurality of records comprising the field set forth in **Table 15** below. This number is analogous to a "version" number for the data stored in the application.

Record description	**External format**		**Internal format(bytes)**	
	Size	Type	Size	Type
Sequence Number	16	AN	16	ASCII

Table 15: Exemplary sequence EF file structure

Card number EF 526 is used to record a unique number identifying the smartcard, and may also be used for key derivation (as described in further detail below). Preferably, card number EF 526 comprises a eight-byte string as set forth in **Table 16** below.

Record description	External format		Internal format(bytes)	
	Size	Type	Size	Type
Card Number	8	HEX	8	HEX

Table 16: Exemplary card number EF

Issuance EF 511 is used to record various details related to the manner in which the application (i.e., cardholder ID DF 406) was created. This file includes information related to the identity of the organization that created the application, as well as information related to the application itself. In a particularly preferred embodiment, issuance EF 511 comprises fields as set forth in **Table 17** below.

Field	External format		Internal format (bytes)	
	Size	Type	Size	Type
Country Authority		ISO 3166	2	
Issuer Authority	10	RID - ISO 7816-5	5	HEX
Application version	5	XX.YY	2	BCD
Application expiration date	8	YYYYMMDD	4	BCD
Application effective date	8	YYYYMMDD	4	BCD
Personalizer Code	1	AN	1	ASCII
Personalization Location	1	AN	1	ASCII

Table 17: Exemplary issuance EF file structure

The personalizer code field shown in **Table 17** refers to the organization that actually "personalizes" the file. That is, before a smartcard may be issued to the cardholder, the database structure must be created within EEPROM 212 (Figure 2), and the initial data values (i.e., default preferences, cardholder name, pin numbers, etc.) must be placed in the appropriate fields within the various EFs. It will be appreciated that, given the nature of the present invention, the smartcard "issuer" and "personalizer" for any given application may not be the same. Therefore, it is advantageous to record various details of the personalization process within smartcard 100 itself. Similar issuance file structures may be provided for the other major applications.

Payment System Application

Referring now to **Figure 6**, payment system application 408 preferably comprises a directory EF 610, issuer DF 602, and a number of optional DFs 603(a)-(n) for use by partnering financial organizations.

Directory EF 610 preferably includes a list of application identifiers and labels as described above in the context of cardholder ID application 406.

Issuer DF 602 comprises pay1 DF 604, which includes data that would traditionally be stored within tracks on a magnetic stripe card (i.e., debit cards, charge cards, and the like). In a preferred exemplary embodiment, pay1 DF 604 comprises a plurality of records having commonly known magnetic-stripe fields as specified in **Table 18** below.

Record descripti n	External f rmat		Internal format(byt s)	
	Size	Type	Size	Typ
Format Code (Track 1)	1	AN	1	ASCII
PAN (Track 2)	15	N	8	BCDF right padding
Expiration date (Track 1 or 2)	4	YYMM	2	BCD
Effective date (Track 1 or 2)	4	YYMM	2	BCD
Discretionary data (Track 1 or 2)	5	N	3	BCDF right padding
Name (Track 1)	26	AN	26	ASCII, LJ blank padding

Table 18: Exemplary Pay1 EF file structure

Airline Application

Referring now to **Figure 7**, airline application 410 preferably comprises directory EF 730, common DF 702, and issuer DF 704, and additional airline applications 703(a), 703(b), and so on.

Directory EF 730 preferably includes a list of application identifiers and labels as described above in the context of cardholder ID application 406.

Common DF 702 generally includes data accessible to all participating airlines, while issuer DF 704 generally includes data which can only be read or written to by the smartcard issuer. Airline application 410 preferably further comprises at least one (preferably three) additional DF 703 for use by airline partnering organizations. That is, one airline partner may have access to and specify the structure of data stored within DF 703(a) (as well as common EF 702), while another airline might have similar access to DF 703(b). These partner DFs preferably conform to the relevant portions of the IATA specification.

Common DF 702 suitably comprises common data which would be of use to any of the various partnering airlines, i.e., passenger EF 706, frequent flier EF 708, IET EF 710, boarding EF 712, and biometric EF 714.

Issuer DF 704, in contrast, comprises information readable by all, but updatable only by the card issuer, i.e., preferences EF 716, PIN EF 718, and issuance EF 720.

Referring now to information stored within common EF 702, passenger EF 706 preferably comprises various records related to the passenger as specified in **Table 19** below.

Record description	**External format**		**Internal format (bytes)**	
	Size	Type	Size	Type
Passenger Name	49	AN	49	ASCII
Gender	1	A	1	BIN
Language Preference	2	AN	2	ASCII
Unique ID	24	AN	24	ASCII
Airline ID (3 letters code)	*3*	*AN*	*3*	*ASCII*
Type code (2 letters)	*2*	*AN*	*2*	*ASCII*
Unique ID	*19*	*AN*	*19*	*ASCII*
Application version	2	N	2	BIN

Table 19: Exemplary passenger EF file structure

In a particularly preferred embodiment, frequent flyer EF 708 comprises a plurality of frequent flier numbers (e.g., ten numbers) having the structure specified in **Table 20** below.

Record descripti n	External format		Internal f rmat (bytes)	
	Size	Type	Siz	Type
Airline Customer ID	22	AN	22	ASCII

Table 20: Exemplary frequent flyer EF file structure

IET EF 710 preferably comprises a plurality of electronic ticket records as set forth in **Table 21** below. The format of these electronic tickets preferably conforms to the IATA standard.

Description of the records	External format		Internal format (bytes)	
	Size	Type	Size	Type
IET 1	14	AN	14	BIN
IET 2	14	AN	14	BIN
IET 3	14	AN	14	BIN
IET 4	14	AN	14	BIN
IET 5	14	AN	14	BIN

Table 21: Exemplary IET file structure

In a particularly preferred embodiment, boarding EF 712 comprises boarding data to be used during check in as specified in **Table 22**. The format of this data preferably conforms to the IATA specification.

Record description	External format		Internal format (bytes)	
	Size	Type	Size	Type
Boarding data	40	AN	40	ASCII

Table 22: Exemplary boarding EF file structure

Biometric EF 714 is suitably used to store biometric data associated with the cardholder, e.g., retina scan data, fingerprint data, or any other sufficiently unique indicia of the cardholder's physical or behavioral characteristics. In a particularly preferred embodiment, biometric EF 714 comprises data as specified in **Table 23** below.

Record description	External format		Internal format (bytes)	
	Size	Type	Size	Type
Biometrics data	100	AN	100	BIN

Table 23: Exemplary biometric EF file structure

Issuance EF 720 is suitably used to hold data related to the issuance of the various applications. In a particularly preferred embodiment, issuance EF 720 comprises a data structure as specified in **Table 24** below.

Field	External format		Internal format (bytes)	
	Size	Type	Size	Type
Country Authority (2 letters)		ISO 3166	2	
Issuer Authority	10	RID - ISO 7816-5	5	HEX
Application version	5	XX.YY	2	BCD
Application expiration date	8	YYYYMM DD	4	BCD
Application effective date	8	YYYYMM DD	4	BCD
Personalizer Code	1	AN	1	ASCII
Personalization Location (custom code)	1	AN	1	ASCII

Table 24: Exemplary issuance EF file structure

PIN EF 718 is suitably used to store PIN values corresponding to each of the participating airline partners. In a particularly preferred embodiment, PIN EF 718 comprises a plurality of records having the structure specified in **Table 25** below, wherein each record is related to the corresponding entry in frequent flyer EF 708 (i.e., record one in EF 718 corresponds to record one in EF 708, and so on.)

Record description	**External format**		**Internal format (bytes)**	
	Size	Type	Size	Type
PIN	8	AN	8	BIN
Expiration date	8	D	4	BCD

Table 25: Exemplary PIN EF file structure

Preferences EF 716, in a particularly preferred embodiment, comprises a preferences array as shown in **Table 26** below. The preference values stored in this file correspond to those discussed below in conjunction with **Table 38**.

Record description	**External format**		**Internal format (bytes)**	
	Size	Type	Size	Type
Preferences Array	8	C	8	BIN

Table 26: Exemplary preferences EF 716 file structure

Rental Car Application

Referring now to **Figure 8**, rental car application 414 preferably comprises common DF 802, directory EF 820, and one or more rental_car DFs 803 (i.e., 803(a), 803(b), and so on) corresponding to individual rental car agencies.

Common DF comprises preferences EF 805, which is described in detail below. Rental_car DFs 803 each comprise a rental_car_id EF 807, reservation EF 809, and expenses EF 811.

Directory EF 820 includes a list of application identifiers and labels for the various DFs under rental_car application 414. The structure of this EF preferably conforms to that described above in the context of cardholder ID application 406.

In a particularly preferred embodiment, preferences EF 805 comprises a set of preferences arrays file structure as shown in **Table 27** below. A preferred list of preference codes for use in each of these arrays is described below in conjunction with **Table 38.**

Record description	External format		Internal format(bytes)	
Preferences Array (Default)	8	C	8	BIN
Preferences Array (No. 2)	8	C	8	BIN
Preferences Array (No. 3)	8	C	8	BIN
Preferred limousine company	12	AN	12	ASCII

Table 27: Exemplary preferences EF

Rental_car_id 807 is used to store frequent rental information, upgrade information, insurance information, and the like. In a particularly preferred embodiment, rental_car_id 807 comprises a file structure as shown in **Table 28** below.

Record description	External format		Internal format(bytes)	
Frequent Rental ID#	22	A	22	ASCII
Company name	*3*	*A*	*3*	*ASCII*
Unique Customer ID	*19*	*A*	*19*	*ASCII*
CDP (Contract Disc. Program)	10	A	10	ASCII
Accumulated points	8	N	3	BIN
Rental features		AR	2	BIN
Car Type Upgrade		*B*	*1 bit*	*B*
Week-end/Vacation Special		*B*	*1 bit*	*B*
Guaranteed Late Reservation		*B*	*1 bit*	*B*
Insurance		Array	2	BIN
Loss Damage Waiver (LDW)		B	1 bit	B
Personal Automobile Insurance		*B*	*1 bit*	*B*
Personal Effects Coverage		*B*	*1 bit*	*B*
Personal Insurance		*B*	*1 bit*	*B*
Corporate Insurance		*B*	*1 bit*	*B*

Table 28: Exemplary rental_car_id EF

Reservation EF 809 is used to store confirmation numbers corresponding to one or more rental car reservations. In a particularly preferred embodiment, reservation EF 809 comprises a plurality of records (e.g., two) having a file structure as shown in **Table 29** below.

Record descripti n	External f rmat		Int rnal format(bytes)	
Rental Car Company	3	A	3	ASCII
Location	3	A	3	ASCII
Date	8	D	4	BCD
Time	4	T	2	BCD
Reservation Number	15	A	15	ASCII
Flight Number	5	M	5	BIN
Airlines	*3*	*AN*	*3*	*ASCII(RJ)*
Flight number	*4*	*N*	*2*	*BCD*
Preferred profile	1	C	1	ASCII

Table 29: Exemplary reservation EF

Expenses EF 811 is used to record expenses incurred by the cardholder during car rental (e.g., the total rental charge). In a particularly preferred embodiment, expenses EF 811 comprises a plurality of records (e.g., five) having a file structure as shown in **Table 30** below.

Record description	External format		Internal format (bytes)	
Type of expense	1	C	1	ASCII
Date	8	D	4	BCD
Location code	3	AN	3	ASCII
Amount	7	N	3	BIN

Table 30: Exemplary expenses EF

Hotel Application

Referring now to **Figure 9**, hotel system application 412 preferably comprises directory EF 920, common DF 914, one or more hotel chain DFs 902, and one or more property DFs 903.

Common DF 914 comprises reservation EF 918, expenses EF 916, key-of-the-room EF 910, and preferences EF 912.

Hotel chain EFs 902(a), 902(b), and so on, comprise preferences EF 904 and stayer ID EF 906 associated with individual hotel chains. In contrast, property EFs 903(a), 903(b), and so on, comprise a similar file structure associated with individual hotel properties (i.e., independent of whether the particular hotel is a member of a nationwide chain).

In a particularly preferred embodiment, reservation EF 918 comprises a plurality of records having the structure shown in **Table 31** below. In general, this EF is used to store confirmation numbers transmitted to smartcard 100 when the cardholder makes a reservation at a given hotel (designated in the property code field). The date field stores the date on which the confirmation number was dispensed.

Record description	External format		Internal format (bytes)	
	Size	Type	Size	Type
Property Code	3	AN	3	ASCII
Date	8	D	4	BCD
Confirmation Number	15	AN	15	ASCII

Table 31: Exemplary reservation EF

Preferences EF 912 preferably comprises three sets of array preferences. The particular codes used in these arrays are discussed below in conjunction with **Table 38**.

Record description	External format		Internal format (bytes)	
	Size	Type	Size	Type
Preferences Array (default)	8	C	8	BIN
Preferences Array (number 2)	8	C	8	BIN
Preferences Array (number 3)	8	C	8	BIN

Table 32: Exemplary preferences EF

Expenses EF 916 preferably comprises a list of recent hotel expenses, for example, room costs, dinner expenses, and the like. In a particularly preferred embodiment, expenses EF 916 comprises a plurality of records (for example, fifteen) arranged in a cyclic file structure and comprising the fields shown in **Table 33** below. Thus, the cardholder is able to examine and print a list of recently incurred expenses by type (a code fixed by convention), date, amount, and property code.

Record description	External format		Internal format (bytes)	
	Size	Type	Size	Type
Type	1	C	1	ASCII
Date	8	D	4	BCD
Property Code	3	AN	3	ASCII
Amount	7	N	3	BIN

Table 33: Exemplary expenses EF

Key-of-the-room EF 910 preferably comprises electronic key values that can be used in conjunction with card readers to provide access to particular hotel rooms. In a particularly preferred embodiment, key-of-the-room EF 910 comprises a plurality of alphanumeric key values as shown in **Table 34** below.

Record description	External f rmat		Internal f rmat (bytes)	
	Size	Type	Size	Type
Key value	40	AN	40	BIN

Table 34: Exemplary key-of-the-room EF

Stayer ID EF 906 preferably comprises frequent stayer data for a particular hotel chain. In a particularly preferred embodiment, Stayer ID EF 906 comprises frequent stayer information as shown in **Table 35** below.

Record description	External format		Internal format (bytes)	
	Size	Type	Size	Type
Frequent stayer number	19	AN	19	ASCII
Frequent Stayer Level Code	1	AN	1	ASCII
Frequent Stayer Level Expiration Date	6	YYYYMM	3	BCD
CDP	10	AN	10	ASCII
Event Counter	3	N	1	BIN
Hotel Frequent Stayer PIN	8	AN	8	BIN

Table 35: Exemplary stayer ID EF

Preferences EF 904 preferably comprises three sets of array preferences as shown in **Table 36**. The particular codes used in these arrays are discussed below in conjunction with **Table 38.**

R c rd descripti n	External format		Internal format (bytes)	
	Size	Type	Size	Type
Preferences Array (default)	8	C	8	BIN
Preferences Array (number 2)	8	C	8	BIN
Preferences Array (number 3)	8	C	8	BIN

Table 36: Exemplary preferences EF

Property DFs 903(a), 903(b), etc., are used in cases where the partnering hotel is not part of a major chain, or when the hotel chooses to employ its own data set independent of its affiliation. In one embodiment, these property DFs are identical in structure to hotel chain DFs 902, except that much of the frequent stayer ID information is removed. More specifically, a typical property DF 903 comprises a preferences EF 938 identical to preferences 904 described above, along with a stayer ID EF 934 which includes only the CDP, event counter, and hotel frequent stayer PIN fields described in conjunction with **Table 33** above. Alternatively, a particular hotel chain or property might choose to implement a different file structure than that described above.

Preference Codes

As mentioned briefly above, a preferred embodiment is configured such that preferences are located in several files distributed throughout smartcard 100; i.e., in preferences EF 514, airline preferences EF 716, hotel preferences EF 912 and 904, and car preferences EF 810. This allows apparently conflicting preferences to coexist within the card depending on context. For example, it is possible to opt for non-smoking in the cardholder ID application while choosing the smoking option within the hotel application. In the case of conflict, preferences are read from the top level to the bottom level, and each level supersedes the previous one.

An exemplary set of codification rules are set forth in **Table 37** below:

0-49	General purpose (Cardholder ID 406)
50-99	Hotel application 412
100-149	Rental car application 414
150-199	Airline application 410
200-255	Other

Table 37: Exemplary Preferences Code Ranges

More specifically, in a preferred exemplary embodiment, preference flags are coded as set forth in **Table 38** below.

Preference	Code (decimal)
GENERAL PURPOSE	
Smoking	00
Non-smoking	01
Home as preferred address	02
Work as preferred address	03
Handicapped	04
Home as preferred e-mail address	05
Work as preferred e-mail address	06
HOTEL PREFERENCES	
King-size bed	50
Queen-size bed	51
Double bed	52
High floor room	53
Low floor room	54
Near elevator room	55
Away from elevator room	56

Preference	Cod (d cimal)
RENTAL CAR PREFERENCES	
Compact car	100
Standard car	101
Mid-size car	102
Luxury car	103
AIRLINE PREFERENCES	
Window seat preferred	150
Aisle seat preferred	151
Low calorie	152
Vegetarian	153
Diabetic	154
Low sodium	155
Kosher	156

Table 38: Exemplary preference codes

Security

In the context of smartcard transactions, data security has five primary dimensions: 1) data confidentiality, 2) data integrity, 3) access control, 4) authentication, and 5) non-repudiation. Each of these dimensions is addressed through a variety of security mechanisms. Data confidentiality, which deals with keeping information secret (i.e., unreadable to those without access to a key), is substantially ensured using encryption technology. Data integrity (and data source verification) focuses on ensuring that data remains unchanged during transfer, and typically employs message authentication techniques. Access control involves card holder verification and other requirements necessary in order for a party to read or update a particular file. Authentication involves ensuring that the card and/or the external device is what it purports to be, and non-repudiation deals with the related task of ensuring that the source of the data or message is authentic, i.e., that a

consumer may not repudiate a transaction by claiming that it was "signed" by an unauthorized party.

Authentication is preferably performed using a "challenge/response" algorithm. In general, authentication through a challenge/response system involves: 1) generation of a random number by a first party; 2) transmission of the random number to a second party (the "challenge", 3) encryption of the random number by the second party in accordance with a key known to both parties, 4) transmission of the encrypted random number to the first party (the "response"), 5) encryption of the random number by the first party, and 6) comparison by the first party of the two resulting numbers. In the case where the two numbers match, authentication is successful; if not, the authentication is unsuccessful. Note that authentication can work both ways: the external world might request authentication of a smartcard (internal authentication), and a smartcard might request authentication of the external world (external authentication). a more detailed account of a preferred challenge/response algorithm can be found in the IBM MFC specification.

In a preferred embodiment, the DES algorithm (Data Encryption Standard) is employed for the various security functions; however, it will be appreciated that any number of other symmetrical or asymmetrical techniques may be used in the context of the present invention. More particularly, there are two general categories of encryption algorithms: symmetric and asymmetric. Symmetric algorithms use the same key for encryption and decryption, for example, DEA (data encryption algorithm) which uses a 56-bit key to encrypt 64-bit blocks of data. Asymmetric algorithms, in contrast, use two different keys: one secret key and one public key. The RSA algorithm, for example, uses two such keys and exploits the computational complexity of factoring very large prime numbers. Additional information these and other cryptographic principles can be found in a number of standard texts, for example: Seberry & Pieprzyk, CRYPTOGRAPHY: AN INTRODUCTION TO COMPUTER SECURITY (1989); Rhee, CRYPTOGRAPHY AND SECURE COMMUNICATIONS (1994); Stinson, CRYPTOGRAPHY: THEORY AND PRACTICE (1995); CONTEMPORARY CRYPTOGRAPHY: THE SCIENCE OF INFORMATION INTEGRITY (1992); and Schneier, APPLIED CRYPTOGRAPHY (2d ed. 1996), the contents of which are hereby incorporated by reference.

Access control is suitably provided by including access conditions within the header of each EF and DF. This prevents a particular operation (e.g., reading or updating) from being performed on a file unless the required access conditions have been fulfilled. Many different access conditions are appropriate in a smart card context. For example, the smartcard might require cardholder verification (i.e., request that the cardholder enter a PIN) before a file operation is allowed. Similarly, internal and/or external authentication as described above might be required.

Another important access condition (referred to herein as the SIGN condition) corresponds to the case where a particular file is "protected" and where updating of a record requires "signing" of the data using a message authentication code (MAC). a MAC can be thought of as a form of electronic seal used to authenticate the content of the message. In a paradigmatic signing procedure, a shortened, encrypted representation of the message (the MAC) is created using a message authentication algorithm (MAA) in conjunction with a key known to both the card and external device. The MAC is then appended onto the message and sent to the card (or external device, depending on context), and the card itself generates a MAC based on the received message and the known key. The card then compares the received MAC with the its own internally-generated MAC. If either the message or MAC was altered during transmission, or the sending party did not use the correct key, then the two MACs will not match, and the access condition will not be fulfilled. If the two MACs correspond, then the access condition is fulfilled, and the particular file operation can proceed.

A MAC may be generated using a variety of MAAs, for example, the ANSI X9.9 method using an eight-byte key, or the ANSI X9.19 method using a sixteen-byte key. Furthermore, the actual key may be "diversified" through encryption with a random number or other appropriate value. These and other details regarding MAC generation can be found in the references cited above as well as the IBM MFC specification.

Two other important access conditions are the NEVER and FREE conditions. The NEVER condition corresponds to the case where a certain file operation (typically updating) is never allowed. The FREE condition, on the other hand, corresponds to

the case where either updating or reading a file record is always allowed, without any additional preconditions for access.

In contrast to the MAC techniques discussed briefly above, non-repudiation is necessarily performed using asymmetrical techniques. That is, as symmetrical techniques such as MAC "sealing" use a key known to more than one party, such techniques can not be used by a third party to ascertain whether the source of the message is correct. Thus, non-repudiation typically employs a public key encryption scheme (e.g., the Zimmerman's PGP system), wherein the sender uses a secret key to "sign" the message, and the receiving party uses the corresponding public key to authenticate the signature. In the context of the present invention, this function is suitably performed by allocating an EF for public and secret key rings, which are well known in the art, along with suitable encryption software resident in the card for assembling the signed message.

Having thus given a brief overview of typical smartcard security procedures, an exemplary set of access conditions is set forth below in **Table 40**. In this regard, the various access conditions for each EF are tabulated with regard to whether the file is being read or updated. In each case, the access condition (FREE, SIGN, etc.), key "owner" (issuer, partner, user, etc.), and key name are listed. In this regard, it will be appreciated that the key name is arbitrary, and is listed here for the sake of completeness.

	READING			UPDATING		
	Access condition	Owner	Key	Access condition	Owner	Key
MF						
DF Cardholder ID 406						
DF Holder_ID 502						
EF ID 504	FREE			SIGN	ISSUER	KEY1
EF Home 506	FREE			SIGN	ISSUER	KEY1
EF Business 508	FREE			SIGN	ISSUER	KEY1
EF Preferences 514	FREE			SIGN	ISSUER	KEY1
EF Passport 516	FREE			SIGN	ISSUER	KEY1
EF Biometrics 522	FREE			SIGN	ISSUER	KEY1
EF Driver 518	FREE			SIGN	ISSUER	KEY1
DF Miscellaneous						
EF Payment card 510	FREE			SIGN	ISSUER	KEY1
EF Sequence 512	FREE			FREE		
EF Card Number 526	FREE			SIGN	ISSUER	KEY1
DF Payment System 408						
DF Issuer 602						
EF Pay1 604	FREE			FREE		
DF Airline 410						
DF Common 702						
EF Passenger 706	FREE			SIGN	ISSUER	KEY2
EF Frequent flier 708	FREE			SIGN	ISSUER	KEY2
EF IET 710	FREE			FREE		
EF Boarding 712	FREE			FREE		
EF Biometric 714	FREE			FREE		
DF Issuer 704						
EF Preferences 716	FREE			SIGN	ISSUER	KEY2
EF PIN 718	FREE			SIGN	ISSUER	KEY2
EF Issuance 720	FREE			SIGN	ISSUER	KEY2
DF Rental car 414						
DF Common 802						
EF Preferences 805	FREE			USER	IDENT	PIN